REMARKS

Claims 1-40 have been replaced with new claims 41-64. The new claims avoid the objections noted in point 3 on page 2 of the Office Action.

Claims 1-8, 11, 12, and 13-33 stand rejected under 35 USC 101 as being directed to non-statutory subject matter. Applicant believes that the new claims 41-64 avoid this rejection.

Claims 45-52 are directed to a software protection arrangement including identifying means, an authorization server and enabling means. Under 35 USC 112, sixth paragraph, a means plus function element covers the corresponding structure, material or acts described in the specification and equivalents thereof. Applicant therefore submits that claims 41-48 are directed to patentable subject matter. See *In re Alappat*, 31 USPQ2d 1545 (Fed. Cir. 1994).

Claims 53-58 are directed to a wireless device and claims 59-61 are directed to a server operable for communication with a wireless device over a wireless network. Applicant submits that such a wireless device and such a server clearly constitute patentable subject matter.

All claims stand rejected over the prior art.

The new independent claims 41, 53 and 59 state that the derived identifier is formed by a function that operates on at least two variables. Thus, these independent claims are directed to the features of the original claims 4, 16 and 22, which were rejected under 35 USC 103 over Hughes et al in view of Mittal et al, Yoshida et al and Yeung et al.

The subject matter of claim 41 is a method of protecting software to be run on a wireless device operable for communication over a wireless network. In accordance with claim 41, an identifier that characterizes the wireless device is created at the wireless device and is transmitted from the wireless device to a server. The server verifies that use of the protected software by the wireless device is authorized and, in response to the verification, executes a predetermined function to form a derived identifier. The predetermined function operates on at least two variables including the identifier received from the wireless device and a decryption key stored by the server. The derived identifier is transmitted from the server to the wireless device and a second predetermined function is

executed at the wireless device on at least two variables including the derived identifier received from the server and the identifier created by the wireless device, to recover a decryption key. The wireless device decrypts the encrypted software using the recovered decryption key to enable execution of the protected software. Successful decryption of the protected software is achieved only in the event that the derived identifier has been formed by the predetermined function operating on the identifier of the wireless device on which the protected software is to be run.

Thus, two conditions must be met in order for the protected software to run on the wireless device. First, the server must verify that use of the protected software is authorized (for example by operating on the identifier with a function and comparing the result returned by the operation with a value stored in the server). Only if the result of the authorization is positive does the server then form the derived identifier, which is sent to the wireless device to recover the decryption key. If the wireless device is able to recover the proper decryption key, the encrypted software is decrypted at the wireless device and can then run on the wireless device.

The subject matter of claim 41 is thus concerned with limiting the circumstances in which software that is present in a wireless device can be run. Suppose, for example, the software is gaming software installed on a mobile phone and the business model chosen by the author of the software is to provide the software at no cost but require the user to pay a small monthly subscription fee to use the software. When the user wishes to use the software to play a game, the enable module 38 (FIG. 1) will attempt to determine from the server whether the user's subscription is up to date and, if so, the software will be decrypted and will run but, if not, the software will not be decrypted and, consequently, will not run.

Hughes et al discloses an arrangement to deter installation of copies of a software program from a CD ROM on multiple machines while not preventing repeated installations of the software program on the first machine on which it is installed. In accordance with the disclosure in Hughes et al, a customer loads the program onto the computer and the program 36 is stored in system memory 22 (paragraph 32). During installation of the software program, the customer enters a CD key 224 (see paragraph 33) and the program 36 stored in system

memory 22 generates a hardware ID that identifies a set of hardware components that make up the customer's computer 20 (paragraph 36). The program concatenates the 15 digit product ID with the 5 digit hardware ID and sends the 20 digit value over a network 336 to an activation server 334 (paragraph 38). The activation server 334 computes a license file from the product ID and the hardware ID using a hashing algorithm. The value of the license file depends on the hardware ID. The activation server stores the product ID, hardware ID and license file in a database 114 (paragraph 40). The activation server 334 returns the license file to the customer computer which stores the license file in the system memory 22 (paragraph 41). When the customer wishes to run the program 36 on the computer 20, the program 36 obtains the product ID and generates the hardware ID and computes a test ID using the same hashing algorithm that the activation server employed to compute the license file. The software program 36 compares the test ID to the license file that was previously provided by the activation server and if the two match, the software program is permitted to operate on the computer whereas if no match occurs, the software program is locked and prevented from operating on the computer (paragraphs 42 and 43). Should the customer attempt to install the program on another computer from the same CD, the program will determine that the test and license files do not match and will prevent operation on the other computer (paragraph 46). Thus, Hughes et al teaches that once the activation server 334 has returned the license file to the customer computer, subsequent verification is performed by the software product 36 itself running on the customer computer, not by the activation server. Hughes et al is not concerned with limiting the circumstances in which a program that has been legitimately installed on a computer can be used but with preventing illicit use of a program on a second (or subsequent) computer after the program has been installed on a first computer. Since Hughes et al makes no reference to either encryption or decryption, Hughes et al necessarily fails to disclose or suggest the step of verifying authorization before attempting to decrypt an encrypted program.

The examiner relies on Mittal et al as teaching the limitation of claim 4 that the predetermined function is a function of at least two

variables, including a received identifier and a confidential decryption key stored at the authorization means.

Mittal et al discloses that encrypted content that is delivered to a computer from a website may be decrypted using a decryption program delivered to the computer system with the encrypted content. In order to guard against copying the downloaded material (encrypted content and decryption program) from the computer to which it was downloaded to another computer, Mittal et al teaches that when a computer requests the delivery of encrypted content from a website, the website requests the computer to return a hash value derived from the computer's processor number and a string that corresponds to the website's URL (FIG. 5, step 510). The website then sends the encrypted content and the decryption program (including the hash value that had been returned by the requesting computer) to the requesting computer. The decryption program periodically generates a hash value from the URL of the website and the processor number of the processor currently executing the program and compares this current hash value with the expected hash value received from the website that supplied the encrypted content and decryption program. If the hash values do not match, the decryption program can discontinue execution. See the paragraph starting at column 2, line 45. Thus, Mittal et al discloses that one may deter unauthorized copying of encrypted video and/or audio content by use of a security measure that employs a hash value derived from a computer's processor number and a website's URL.

Mittal et al does not disclose or suggest the limitation of claim 41 that the predetermined function that operates on at least two variables is executed at the server. On the contrary, Mittal et al indicates that the microprocessor 203, which generates the hash value 204 from the key 201 and the identifier 205 is a component of the computer to which the encrypted content and accompanying decryption program are downloaded and is not part of an authorization server. Further, Mittal et al does not suggest that one of the variables used to create the hash value is a decryption key stored by a server at the website.

The examiner relies on Yoshida et al as teaching recovering a confidential decryption key for use as a decryption key in decrypting encrypted code.

FIG. 1 of Yoshida et al discloses a software distribution system by which a software vendor 14 distributes software 100 on CD ROM 10 to personal computers 11 each including a hard disk device 12 containing a decryption key memory unit 13. As stated at column 6, lines 3-11, the decryption key memory unit is a data file that stores the software ID of each encrypted software 100 that has once been installed and the corresponding decryption key for decrypting the encrypted software 100.

The teaching of Yoshida does not cure the deficiencies in the disclosure of Hughes et al. Specifically, Yoshida et al does not disclose or suggest that a decryption key is stored separately from the personal computer 11 and is used to form a derived identifier that is sent to the personal computer 11.

The examiner asserts that column 6, lines 10-20 of Yeung teach execution of a second predetermined function of at least two variables, including the identifier and the derived identifier, to recover the confidential decryption key.

The passage starting at column 6, line 4, of Yeung relates to an embodiment where visual/perceptual scrambling and data scrambling can be performed in the alternative. Evidently, one key is associated with visual/perceptual scrambling and the other with data scrambling. See column 5, line 66 to column 6, line 3. The keys 251 and 252 are transformations of data or portions thereof of client-based information 185, i.e. information specific to a client platform (column 6, lines 18-19). Column 6, lines 10-19, of Yeung et al suggest that the key 251 and the key 252 are each derived from a mapping or transformation of any one of the CPU_ID, the REC_ID, the concatenated sum of CPU_ID and auxiliary information, and a hash result of CPU_ID.

Claim 41 is quite specific in requiring that the decryption key be stored by the server. Yeung et al, however, appears to show that the mapping function generates the keys from client-based information. The keys produced by the mapping function 255 are used for encryption, not decryption. Accordingly, applicant submits that the teaching in Yeung et al does not cure the deficiencies in the disclosure of Hughes et al.

In view of the foregoing, applicant submits that the subject matter of claim 41 is not disclosed or suggested by Hughes et al,

Mittal et al, Yoshida et al and Yeung et al, whether taken singly or in combination. Therefore, claim 41 is patentable and it follows that the dependent claims 42-44 also are patentable.

Claim 45 is directed to a software protection arrangement including identifying means, an authorization server and enabling means. The functions attributed to the authorization server and the enabling means correspond to features that are discussed above in connection with claim 41. For the reasons discussed in connection with claim 41, applicant submits that claim 45 is patentable and it follows that the dependent claims 46-52 and 62 also are patentable.

The subject matter of claim 53 is a wireless device including identifying means and enabling means. The enabling means is operable to receive a derived identifier and to decrypt the encrypted software using the recovered decryption key provided that the derived identifier has been formed by a predetermined function operating on the identifier of the wireless device.

As noted above, Hughes et al makes no reference to either encryption or decryption and therefore does not disclose the function of the enabling means of claim 53. Claim 53 requires that the enabling means receive a derived identifier formed by a predetermined function operating on at least two variables including the identifier transmitted by the wireless device and the decryption key. Mittal et al does not disclose or suggest that a predetermined function that operates on at least two variables to generate a derived identifier provided to enabling means operates elsewhere than on the computer to which the encrypted content and accompanying decryption program are downloaded. Further, Mittal et al does not suggest that one of the variables used to create the hash value is a decryption key stored by a server at the website from which the encrypted content is downloaded.

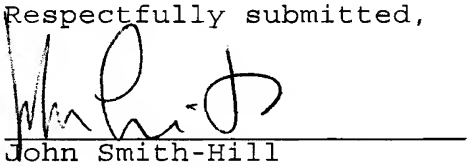
As discussed above, Yoshida et al does not disclose or suggest that a decryption key is stored separately from the personal computer 11 and is used to form a derived identifier that is sent to the personal computer and can be operated on to recover the decryption key.

For similar reasons to those discussed above in connection with claim 41, applicant submits that the teaching in Yeung et al does not cure the deficiencies in the disclosure of Hughes et al.

In view of the foregoing, applicant submits that the subject matter of claim 53 is not disclosed or suggested by the cited references, whether taken singly or in combination. Therefore, claim 53 is patentable and it follows that the dependent claims 54-58 and 63 also are patentable.

The subject matter of claim 59 is a server operable for communication with a wireless device over a wireless network. Claim 59 specifies that the server is operable to perform a verification step similar to that specified in claim 41 and to execute a predetermined function to form a derived identifier, in the same manner as the method of claim 41. For the reasons discussed above in connection with these steps of claim 41, applicant submits that the subject matter of claim 59 is not disclosed or suggested by the cited references, whether taken singly or in combination. Therefore, claim 59 is patentable and it follows that the dependent claims 60, 61 and 64 also are patentable.

Respectfully submitted,



John Smith-Hill
Reg. No. 27,730

SMITH-HILL & BEDELL, P.C.
16100 N.W. Cornell Road, Suite 220
Beaverton, Oregon 97006

Tel. (503) 574-3100
Fax (503) 574-3197
Docket: FORR 2793